Sensitivity to synchronism in some boolean automata networks

Marco Montalva Medel, Kévin Perrot, Pedro Oliveira, Eurico Ruivo

Sensitivity to synchronism in some Boolean automata networks

M. Montalva-Medel[1], K. Perrot[2], P.P.B. de Oliveira[3], and E.P. Ruivo[3]

[1] FIC, Universidad Adolfo Ibáñez, Santiago, Chile
[2] Université d'Aix-Marseille, CNRS, LIF (UMR 7279), Marseille, France
[3] Universidade Presbiteriana Mackenzie, São Paulo, Brazil
marco.montalva@uai.cl

Abstract. We study the sensitivity of some Boolean automata networks to changes in their dynamics against deterministic update perturbations. Due to their large number of different dynamics, they can be extremely sensitive to update schedule perturbations, which renders them not robust in this sense, a feature often undesirable in many applications. Here, we study the maximum number of different dynamics in elementary cellular automata, with fixed, cyclic lattices. First, we formally prove the estimate $3^n + 2 - 2^{n+1}$ for such a number, empirically proposed in a previous work, as well as its sharpness, by proving that some rules actually reach it. Finally, we discuss possible key follow-ups to the present study.

1 Introduction

Given a Boolean automata network (BAN) with n nodes, the determination of all its different dynamics out of deterministic update schedules can be a computationally intensive process, since both the number of deterministic update schedules and the number of configurations grow exponentially as n increases. Given that these networks may display a large number of different dynamics, they can be extremely sensitive to update schedule perturbations, which is a form of lack of robustness, often undesirable in many applications.

Nevertheless, in [3] an upper bound was established for the number $|D(G)|$ of different dynamics (details in Section 2), that depends only on the interaction digraph G, and involves the concept of update digraph (introduced in [4]) and the set $U(G)$ that groups them. Put it simply, an update digraph defines for each arc whether the tail is updated before or after the head of the arc. This information is sufficient to completely define the dynamics of a BAN [4]. In this context, it is an open problem to determine a mathematical expression for $|D(G)|$, although some exact formulas do exist for $|U(G)|$ when the interaction digraph has particular topologies [3, 2, 1]. In [5], computational experiments were presented that allowed to observe two important facts for elementary cellular automata (ECAs) with fixed cyclic lattice size n; the number $|U(G)|$ evolves as $3^n + 2 - 2^{n+1}$ and about 57% of the 256 ECA rules showed that $|D(G)| = |U(G)| = 3^n + 2 - 2^{n+1}$. This paper is the starting point to demonstrate the two previous facts; for the first one, we established the (main) Theorem 3 that formally prove it and, for the second fact, still in-progress, we give its current status.

2 Definitions, notations and preliminary results

Definition 1. *A Boolean automata network (BAN) $\mathcal{N}$ of size n is defined by a set of n local transition functions $\{f_i : \{0,1\}^n \to \{0,1\}\}_{0\leq i\leq n-1}$, one for each automaton of the network. A configuration x is an element of $\{0,1\}^n$ which gives the Boolean state of each automaton.*

Let $x = (x_0, \ldots, x_{n-1}) \in \{0,1\}^n$ be a Boolean vector of size n, we denote $\overline{x}^i$ the vector such that $\overline{x}^i_j = x_j$ for all $j \neq i$, and $\overline{x}^i_i = 1 - x_i$.

Definition 2. *The interaction digraph of a BAN $\mathcal{N}$ is the digraph $G = (V, A)$, with $V = \{0, \ldots, n-1\}$, and such that $(i, j) \in A$ if and only if there exists x such that $f_j(\overline{x}^i) \neq f_j(x)$ (i.e., i has an influence on j).*

Definition 3. *An update schedule s is a function $s : \{0, \ldots, n-1\} \to \{0, \ldots, n-1\}$ telling the order in which the automata are updated. To begin, every automaton i such that $s(i) = 0$ has its Boolean state updated according to x, yielding y, then every automaton i such that $s(i) = 1$ is updated according to y,* etc. *We denote $F^s(x)$ the image of configuration x under update schedule s.*

Definition 4. *Given an interaction digraph $G = (V, A)$ and an update schedule s, we define the label function $lab_s : A \to \{\ominus, \oplus\}$ as follows:*

$$\forall (i,j) \in A,\ lab_s(i,j) = \begin{cases} \oplus & \text{if } s(i) \geq s(j) \\ \ominus & \text{if } s(i) < s(j). \end{cases}$$

Definition 5. *The update digraph of G with update schedule s is defined as the labeled digraph (G, lab_s). We denote by $U(G)$ the set of update digraphs associated to G.*

The above concept was introduced in [4] where the following result was proved in order to group equal dynamics:

Theorem 1. *Let $(\mathcal{N}, s_1)$ and $(\mathcal{N}, s_2)$ be two BANs, with interaction graph G, that differ only in the update schedule. If $(G, lab_{s_1}) = (G, lab_{s_2})$, then $(\mathcal{N}, s_1)$ and $(\mathcal{N}, s_1)$ have the same dynamics.*

Definition 6. *Given an interaction digraph G of a BAN $\mathcal{N}$, we define $D(G)$ as the set of dynamics (transition graphs) of $\mathcal{N}$ obtained with every (deterministic) update schedule.*

As a direct consequence of Theorem 1, we have the following result that establishes an upper bound for the number of different (deterministic) dynamics of a BAN $\mathcal{N}$.

Corollary 1. $|D(G)| \leq |U(G)|$

Definition 7. *An Elementary Cellular Automaton (ECA) is a BAN with interaction digraph in the form of a circle digraph, with all its local functions being the same $f : \{0,1\}^3 \to \{0,1\}$, $(x_{i-1}, x_i, x_{i+1}) \to f(x_{i-1}, x_i, x_{i+1})$. Function f is sometimes referred to as Wolfram rule r (or simply, rule r), with $r \in \{0, ..., 255\}$, since there are $2^8 = 256$ possible ECAs.*

3 Bounding $|U(G)|$ for ECAs

The interaction digraph $G = (V, A)$ of an ECA of size n is composed of n vertices $V = \{0, \dots, n-1\}$ and $2n$ arcs $A = \{(i, i+1 \mod n) \mid i \in V\} \cup \{(i+1 \mod n, i) \mid i \in V\}$, as depicted in Figure 1.

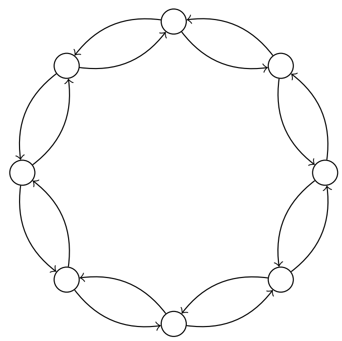


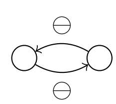
Fig. 1. Interaction digraph of an ECA of size 8. Note that some ECA may have only a subset of the depicted arcs (for example rule 0 has no arc, and rule 1 has all these arcs).

Theorem 2 ([3]). *($Glab_s$) is an update digraph if and only if the same graph where the orientation of all negative arcs is reversed does not contain any negative cycle.*

Theorem 3. *For any ECA, it holds that $|U(G)| \leq 3^n + 2 - 2^{n+1}$, and the bound is tight.*

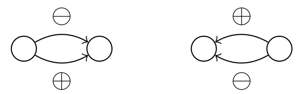
Proof. Let us consider the valid labelings of the interaction digraph of an ECA.

– According to Theorem 2, the pattern



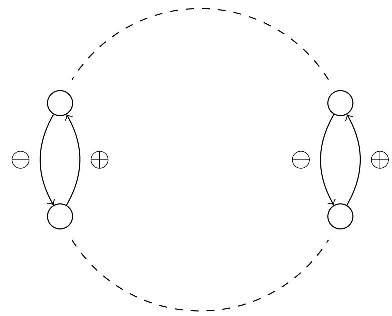
is forbidden and, as a consequence, there are three possibilities for each cycle of size two: ⊕⊕, ⊕⊖ and ⊖⊕ (but not ⊖⊖); hence 3^n possibilities of labels for the whole interaction digraph so far (without considering the combinations creating forbidden patterns).

– Forbidden cycles of Theorem 2 are of length two or n for ECAs, because if the cycle is ⊖⊕ or ⊕⊖, then one of the two following subgraphs is created when the orientation of minus arcs are reversed (notice that if no cycle exists, then the update digraph has all ⊕⊕ and is valid):



This prevents any negative cycle that does not make a whole tour around the graph (that is, a negative cycle around the n vertices).
As a consequence, the two following lemmas hold.

Lemma 1. *If the pattern P below appears in the update digraph, then there are no forbidden patterns for Theorem 2.*

$P =$  .

Lemma 2. *If the pattern P does not appear in the update digraph and at least one $\oplus\ominus$ or one $\ominus\oplus$ is present, then there exists a forbidden pattern for Theorem 2.*

- Thanks to the two above lemmas we can count, among the 3^n possibilities of labels, which ones create forbidden patterns:
  1. Choose an orientation for the $\ominus$ (either clockwise or counter-clockwise), because they all have to point in the same direction.
  2. Then choose which of the n cycles of size two are $\ominus\oplus$ (or $\oplus\ominus$ for the other orientation), and which are $\oplus\oplus$. Here, at least one $\ominus\oplus$ (respectively $\oplus\ominus$) is required; hence the choice all $\oplus\oplus$ is discarded.
- This leads to $2(2^n - 1)$ possibilities of creating forbidden patterns.
- Finaly, we get $3^n - 2(2^n - 1) = 3^n + 2 - 2^{n+1}$ valid update digraphs.

4 ECAs for which $|D(G)| = |U(G)|$

The general idea is to prove that, given two update schedules $s_1 \neq s_2$, we can construct a configuration $x \in \{0, 1\}^n$ (at least for $n > 4$) such that $F^{s_1}(x) \neq F^{s_2}(x)$, which implies that $|D(G)| = |U(G)|$.

4.1 ECA rule 1

Let s_1 and s_2 be two update schedules such that $lab_{s_1}(i, i+1) = \oplus$ and $lab_{s_2}(i, i+1) = \ominus$ for some i.

The goal is to find a configuration $x \in \{0, 1\}^n$ such that

- in s_1 we have $x_i = 0 \wedge x_{i+1} = 0 \wedge x_{i+2} = 0$ at "time" $s_1(i+1)$, which implies $x_{i+1} \mapsto 1$;
- in s_2 we have $x_i = 1$ at "time" $s_2(i+1)$, which implies $x_{i+1} \mapsto 0$,

so that $F^{s_1}(x) \neq F^{s_2}(x)$.

- Part $\geq i+1$:
  - Constraints given by s_1:
    - $x_{i+1} = 0$
    - $x_{i+2} = \begin{cases} 0 & \text{if } lab_{s_1}(i+2, i+1) = \oplus \\ 1 & \text{if } lab_{s_1}(i+2, i+1) = \ominus \end{cases}$

    $\implies x_{i+1} = 0 \wedge x_{i+2} = 0$ at "time" $s_1(i+1)$.
- Part $\leq i$:
  - Constraints given by s_1:
    - $x_i = 0 \implies x_i = 0$ at "time" $s_1(i+1)$.
  - Constraints given by s_2:
    - $x_i = 0$ (to comply with s_1)
    - $x_{i-1} = \begin{cases} 0 & \text{if } lab_{s_2}(i-1, i) = \oplus \\ 1 & \text{if } lab_{s_2}(i-1, i) = \ominus \end{cases}$

    $\implies x_i = 1$ at "time" $s_2(i+1)$.

4.2 ECA rule 2

Let s_1 and s_2 be two update schedules such that $lab_{s_1}(i, i+1) = \oplus$ and $lab_{s_2}(i, i+1) = \ominus$ for some i.

The goal is to find a configuration $x \in \{0, 1\}^n$ such that

- in s_1 we have $x_i = 1$ at "time" $s_1(i+1)$, which implies $x_{i+1} \mapsto 0$;
- in s_2 we have $x_i = 0 \wedge x_{i+1} = 0 \wedge x_{i+2} = 1$ at "time" $s_2(i+1)$, which implies $x_{i+1} \mapsto 1$,

so that $F^{s_1}(x) \neq F^{s_2}(x)$.

- Part $\geq i+1$:
  - Constraints given by s_2:
    - $x_{i+1} = 0$
    - $x_{i+2} = \begin{cases} 1 & \text{if } lab_{s_2}(i+2, i+1) = \oplus \\ 0 & \text{if } lab_{s_2}(i+2, i+1) = \ominus\text{, and } x_{i+3} = \begin{cases} 1 & \text{if } lab_{s_2}(i+3, i+2) = \oplus \\ 0 & \text{if } lab_{s_2}(i+3, i+2) = \ominus\dagger \end{cases} \end{cases}$

      † etc, having all $\ominus$ is impossible so it eventually stops and we get a configuration.

    $\implies x_{i+1} = 0 \wedge x_{i+2} = 1$ at "time" $s_2(i+1)$.
- Part $\leq i$:
  - Constraints given by s_1:
    - $x_i = 1 \implies x_i = 1$ at "time" $s_1(i+1)$.
  - Constraints given by s_2:
    - $x_i = 1$ (to comply with s_1) $\implies x_i = 0$ at "time" $s_2(i+1)$.

so that $F^{s_1}(x) \neq F^{s_2}(x)$.

- Part $\geq i+1$:
  - Constraints given by s_2:
    - $x_{i+1} = 0$
    - $x_{i+2} = \begin{cases} 1 & \text{if } lab_{s_2}(i+2, i+1) = \oplus \\ 0 & \text{if } lab_{s_2}(i+2, i+1) = \ominus, \text{ and } x_{i+3} = \begin{cases} 1 & \text{if } lab_{s_2}(i+3, i+2) = \oplus \\ 0 & \text{if } lab_{s_2}(i+3, i+2) = \ominus \dagger \end{cases} \end{cases}$

      † etc, having all $\ominus$ is impossible so it eventually stops and we get a configuration.

    $\implies x_{i+1} = 0 \wedge x_{i+2} = 1$ at "time" $s_2(i+1)$.
- Part $\leq i$:
  - Constraints given by s_1:
    - $x_i = 1 \implies x_i = 1$ at "time" $s_1(i+1)$.
  - Constraints given by s_2:
    - $x_i = 1$ (to comply with s_1) $\implies x_i = 0$ at "time" $s_2(i+1)$.

Let s_1 and s_2 be two update schedules such that $lab_{s_1}(i+1, i) = \oplus$ and $lab_{s_2}(i+1, i) = \ominus$ for some i.

The goal is to find a configuration $x \in \{0, 1\}^n$ such that

- in s_1 we have $x_{i-1} = 0 \wedge x_i = 0 \wedge x_{i+1} = 1$ at "time" $s_1(i)$, which implies $x_i \mapsto 1$,
- in s_2 we have $x_{i+1} = 0$ at "time" $s_2(i)$, which implies $x_i \mapsto 0$;

so that $F^{s_1}(x) \neq F^{s_2}(x)$.

- Part $\leq i$:
  - Constraints given by s_1:
    - $x_i = 0$
    - $x_{i-1} = 0$

    $\implies x_{i-1} = 0 \wedge x_i = 0$ at "time" $s_1(i)$.
- Part $\geq i+1$:
  - Constraints given by s_1:
    - $x_{i+1} = 1 \implies x_{i+1} = 1$ at "time" $s_1(i)$.
  - Constraints given by s_2:
    - $x_{i+1} = 1$ (to comply with s_1) $\implies x_{i+1} = 0$ at "time" $s_2(i)$.

4.3 ECA rule 110

Let s_1 and s_2 be two update schedules such that $lab_{s_1}(i, i+1) = \oplus$ and $lab_{s_2}(i, i+1) = \ominus$ for some i.

The goal is to find a configuration $x \in \{0, 1\}^n$ such that

- in s_1 we have $x_i = 1 \wedge x_{i+1} = 1 \wedge x_{i+2} = 1$ at "time" $s_1(i+1)$, which implies $x_{i+1} \mapsto 0$,
- in s_2 we have $x_i = 0 \wedge x_{i+1} = 1 \wedge x_{i+2} = 1$ at "time" $s_2(i+1)$, which implies $x_{i+1} \mapsto 1$;

so that $F^{s_1}(x) \neq F^{s_2}(x)$.

- Part $\geq i+1$:
  - Constraints given by s_1 and s_2:
    * $x_{i+1} = 1$
    * $x_{i+2} = 1 \wedge x_{i+3} = 0 \wedge x_{i+4} = 0 \wedge x_{i+5} = 0 \wedge \cdots \wedge x_{i+j} = 0$ while $lab_{s_1}(i+j+1, i+j) = \ominus$ or $lab_{s_2}(i+j+1, i+j) = \ominus$ (and $x_{i+j+1} = 0$ for $\max\{\min\{j \mid lab_s(i+j+1, i+j) = \oplus\} \mid s = s_1, s_2\}$).

    $\Longrightarrow x_{i+2} = 1$ at "time" $s_1(i+1)$ and $s_2(i+1)$.
- Part $\leq i$:
  - Constraints given by s_1:
    * $x_i = 1$
  - Constraints given by s_2:
    * $x_i = 1$ (to comply with s_1)
    * we need $x_{i-1} = 1$ at "time" $s_2(i)$

      $x_{i-1} = \begin{cases} 1 & \text{if } lab_{s_2}(i-1, i) = \oplus \\ 0 & \text{if } lab_{s_2}(i-1, i) = \ominus \end{cases}$

Let s_1 and s_2 be two update schedules such that $lab_{s_1}(i+1, i) = \oplus$ and $lab_{s_2}(i+1, i) = \ominus$ for some i.

5 Discussion and perspectives

We have formally proved in the (main) Theorem 3 the estimation of $|U(G)|$ for ECAs proposed in [5] and also proved that it is tight, by showing some ECA rules that reach such a bound. Is quite obvious that ineffective links imply that $|D(G)| < 3^n + 2 - 2^{n+1}$ because the 'real' bound for $|U(G)|$ is strictly smaller; but what causes such a decrease? The answer is not trivial and intuition might suggest that the presence of fixed points somehow 'kills' the sensitivity to update schedule, because they remain the same, whatever the update schedule; however, this is not the case. In fact, ECA rule 73, for $n = 4k$, is such that it

1. has at least one fixed point, namely, $(0011)^k$; and
2. reaches the bound $|D(G)| = |U(G)|$ (at least for $n \leq 4$).

On a different perspective, we know that the influence of the link effectiveness is a necessary condition; but, is it sufficient? Again, the answer is negative, since ECA rule 8, for $n \leq 4$,

1. has all effective links (hence $|U(G)| = 3^n + 2 - 2^{n+1}$); and
2. does not reach the bound, i.e., $|D(G)| < |U(G)|$.

All above gives insights that drives us to continue with the study of the number of different dynamics in ECAs; in fact, we are presently working on the proofs of the remaining ECA rules not referred to in the present paper.

6 Acknowledgments

This work was supported by FONDECYT Iniciación 11150827 [M.M-M.], FONDECYT Grant 3140527 (DIM, Universidad de Chile), Núcleo Milenio Información y Coordinación en Redes (ACGO) [K.P.], MackPesquisa (Fundo Mackenzie de Pesquisa) and CNPq [P.P.B.O.].

References

1. J Aracena, J Demongeot, E Fanchon, and M Montalva. On the number of different dynamics in boolean networks with deterministic update schedules. *Mathematical biosciences*, 242(2):188–194, 2013.
2. Julio Aracena, Jacques Demongeot, Eric Fanchon, and Marco Montalva. On the number of update digraphs and its relation with the feedback arc sets and tournaments. *Discrete Applied Mathematics*, 161(10):1345–1355, 2013.
3. Julio Aracena, Eric Fanchon, Marco Montalva, and Mathilde Noual. Combinatorics on update digraphs in boolean networks. *Discrete Applied Mathematics*, 159(6):401–409, 2011.
4. Julio Aracena, Eric Goles, Andrés Moreira, and Lilian Salinas. On the robustness of update schedules in boolean networks. *Biosystems*, 97(1):1–8, 2009.
5. Marco Montalva-Medel, P.B. de Oliveira, and Eric Goles. On the maximum number of different dynamics in elementary cellular automata. In *2016 Conference on Complex Systems (CCS'16), Beurs Van Berlage, Amsterdam, The Netherlands, 19-22 September*, 2016.